Automatic Learning Assistant in Telugu

Meghana Bommadi, Shreya Terupally, Radhika Mamidi
Language Technologies Research Centre
International Institute of Information Technology Hyderabad, India
meghana.bommadi@research.iiit.ac.in , shreya.reddy@students.iiit.ac.in,
radhika.mamidi@iiit.ac.in

Abstract

This paper presents a learning assistant that tests one's knowledge and gives feedback that helps a person learn at a faster pace. A learning assistant (based on an automated question generation) has extensive uses in education, information websites, self-assessment, FAQs, testing ML agents, research, etc. Multiple researchers, and companies have worked on Virtual Assistance, but majorly in English. We built our learning assistant for Telugu language to help with teaching in the mother tongue, which is the most efficient way of learning[1]. Our system is built primarily based on Question Generation in Telugu.

Many experiments were conducted on Question Generation in English in multiple ways. We have built the first hybrid machine learning and rule-based solution in Telugu, which proves efficient for short stories or short passages in children's books. Our work covers the fundamental question forms with question types: adjective, yes/no, adverb, verb, when, where, whose, quotative, and quantitative (how many/how much). We constructed rules for question generation using Part of Speech (POS) tags and Universal Dependency (UD) tags along with linguistic information of the surrounding relevant context of the word. Our system is primarily built on question generation in Telugu, and is also capable of evaluating the user's answers to the generated questions.

1 Introduction

Research on Virtual Assistants is renowned since they being widely used in recent times for numerous tasks. These assistants are generated using large datasets and high-end Natural Language Understanding (NLU) and Natural Language Generation (NLG) tools. NLU and NLG are used in interactive NLP applications such as AI-based dialogue systems/voice assistants like SIRI, Google Assistant, Alexa, and similar personal assistants. Research is still going on to make these assistants work in major Indian languages as well.

An automated learning assistant like our system is not only useful for the learning process for humans but also for machines in the process of testing ML systems[2]. Research has been done for Question Answer generating system in English[3], concentrating on basic Wh-questions with a rule-based approach[4], question template based approaches[5] etc. For a low-resourced language like Telugu, a complete AI-based solution can be non-viable. There are hardly any datasets available for the system to produce significant accuracy. A completely rule-based system might leave out principle parts of the abstract. There is a chance that all the questions cannot be captured inclusively by completely handwritten rules. Hence, we want to introduce a mixed rule-based and AI-based solution to this problem.

Our system works on the following three crucial steps:

1. Summarization
2. Question Generation
3. Answer Evaluation

We implemented summarization using two techniques viz. Word Frequency (see 4.1), and TextRank (see 4.2) which are explained further in section 4. Summarization

We attempted to produce questions, concentrating on the critical points of a text that are generally

[1] (Roshni, 2020) (Nishanthi, 2020)

[2] (Hidenobu Kunichika, 2004)

[3] (Maria Chinkina, 2017)

[4] (Payal Khullar)

[5] (Hafedh Hussein, 2014)

Proceedings of the 1st Workshop on Document-grounded Dialogue and Conversational Question Answering, pages 29–37
August 5–6, 2021. ©2021 Association for Computational Linguistics

asked in assessment tests. Questions posed to an individual challenge their knowledge and understanding of specific topics, so we formed questions in each sentence in as many ways as possible. We based this model on children's stories, so the questions we wanted to produce aim to be simpler and more objective.

Based on the observation of the data chosen and analysis of all the possible causes, we developed a set of rules for each part of speech that can be formed into a question word in Telugu. We maximized the possible number of questions in each sentence with all the keywords. We built rules for question generation based on POS tags, UD tags and information surrounding the word, which is comparable with *Vibhaktis* (case markers) in Telugu grammar.

The Question Generation in manually evaluated and the detailed error analysis is given in section 8.1. Our Learning Assistant evaluates using string matching, keyword matching for Telugu answers, and a pre-trained sentence transformer model using XLM-R.(Nils Reimers, 2019)

2 Related Work

Previously, Holy Lovenia, Felix Limanta et al.[2018] (Holy Lovenia, 2018) experimented on Q&A pair Generation in English where they succeeded in forming What, Who, and Where questions. Rami Reddy et al.[2006] (Rami Reddy Nandi Reddy, 2006) worked on Dialogue based Question Answering System in Telugu for Railway inquiries, which majorly concentrated on Answer Generation for a given query. Similar work has done by (Hoojung Chung) on dealing with practical question answering system in restricted domain. Shudipta Sharma et al.[2018](Shudipta Sharma) worked on automatic Q&A pair generation for English and Bengali texts using NLP tasks like verb decomposition, subject auxiliary inversion for a question tag.

3 Dataset

We have used a Telugu stories dataset taken from a website called kathalu wordpress".[6] This dataset was chosen because of a variety in the themes of the stories, wide vocabulary and sentences of varying lengths.

1. Number of stories : 21
2. Average number of sentences : 56
3. Average number of words : 281
4. Genre of the stories : Moral Stories for Children

For testing we used stories by Prof. N. Lakshmi Aiyar:

1. Number of stories : 5
2. Average number of sentences : 190
3. Average number of words : 1060
4. Genre of the stories : Realistic Fiction

4 Summarization

Since Telugu is a low resource language, we used statistical and unsupervised methods for this task. Summarization also ensures the portability of our system to other similar low resource languages.

For summarization, we did a basic data preprocessing (spaces, special characters, etc.) in addition to root-word extraction using Shiva Reddy's POS tagger[7].

We used two types of existing summarization techniques:

1. Word Frequency-based summarization
2. TextRank based frequency

4.1 Word Frequency-based Summarization

WFBS (Word Frequency-based Summarization) is calculated using the word frequency in the passage.[8] This process is based on the idea that the keywords or the main words will frequently appear in the text, and those words with lower frequency have a high probability of being less related to the story.

All the sentences that carry crucial information are produced successfully by this method because the keywords are used repeatedly in children's stories, subsequently causing the highest frequency.

We used a dynamic ratio (a ratio that can be changed or chosen by the user as an input) for getting the desirable amount of summary (short summary or a longer summary, for example: k% of

[6] https://kathalu.wordpress.com/

[7] http://sivareddy.in/downloads

[8] (Ani Nenkova) (Mr. Shubham Bhosale)

the sentences, the system will output k% of sentences with the highest frequent words from the dictionary) This ratio, when dynamically changed, performed better than the fixed ratio of word selection.

Steps followed in WFBS are:

1. Sentences are extracted from the input file.
2. The file is prepossessed and the words are tokenized.
3. Stop words are removed.
4. Frequency of each word is calculated and stored in dictionaries.
5. The sentences with least frequent word are removed.
6. Calculated the ratio of words that occur in highest to lowest frequency order.

4.2 TextRank based Frequency

TextRank is a graph-based ranking model[9] that prioritizes each element based on the values in the graph. This process is done in the following steps:

1. A graph is constructed using each sentence as a node
2. Similarity between the two nodes is marked as the edge weight between the nodes
3. Each sentence is ranked based on the similarity with the whole text
4. The page-rank algorithm is run until convergence
5. The sentences with top N ranking as summarized text is given as the output

The TextRank algorithm is a graph based method that updates the sentence score WS iteratively using the following equation (1).

$$WS(V_i) = (1-d) + d * \sum_{V_i \varepsilon In(V_i)} \frac{w_{ij}}{\sum_{V_k \varepsilon Out(V_j)} w_{jk}} WS(V_j) \tag{1}$$

Where d = damping factor (0.85), w_{ij} is the similarity measure between ith and jth sentences.

This method has the advantage of using the similarity between the two sentences to rank them instead of high-frequency words.

We used two kinds of **similarity** measures for the TextRank based summarization:

1. **Common words**: A measure of similarity based on the number of common words in two sentences after removing stop words. We used root word extraction of the common words for better results since Telugu is a fusional and agglutinative language and has repeated words with a different suffix each time.

2. **Best Match 25**: A measure of the similarity between two passages, based on term frequencies in the passage.[10]

The results observed by this method captures crucial information of the story, but lesser readability and fluency was observed. Within the similarity measures, BM25 has shown slightly better results since the BM25 algorithm ranks sentences based on the importance of particular words (inverse document frequency - IDF) instead of just using the frequency of words.

5 Answer Phrase Selection

Candidate answers are words/phrases that depict some vital information in a sentence. Adjectives, adverbs, and the subject of a sentence are some examples of such candidates.

The answer selection module utilizes two main NLP components - POS Tagging (Part of Speech tagging) and UD parsing (Universal Dependency parsing), along with language-specific rules to determine the answer words in an input sentence.

5.1 POS Tagging

We followed state-of-the-art method by Siva Reddy et al. (2011) (Siva Reddy, 2011), Cross-Language POS Taggers" an implementation of a TnT-based Telugu POS Tagger [11] to parse our data.

The tagger learns morphological analysis and POS tags at the same time, and outputs the lemma (root word), POS tag, suffix, gender, number and case marker for each word.

The model was pre-trained on a Telugu corpus containing approximately 3.5 million tokens and

[9] (Joshi, 2018)(Liang, 2019)

[10] (Federico Barrios, 2016)

[11] https://bitbucket.org/sivareddyg/telugu-part-of-speech-tagger/src/master/

had an evaluation accuracy of 90.73% for the main POS tag.

5.2 UD Tagging

A Bi-LSTM model using Keras is structured and trained using Telugu UD tags dataset UD_Telugu-MTG". [12]

The Bi-LSTM model outputs the UD tags for each word in a sentence using Keras. We considered the subject, which is marked subj" by UD tagger, as the selected answer phrase for a sentence based on the condition that it marked root and punctuation correctly.

This model gave 85% accurate results, including the PAD tags(padding tags), which might not be an adequate result, but based on the conditions and given that the tags subj" is labeled in a sentence scarcely, the results have been considered to be acceptable.

5.3 Rules

The outputs of the POS Tagging and UD Parsing modules are used as the crucial markers in our language-specific rules. In addition to conditions based on word surroundings, these tags select one or more answer phrases in each sentence.

We classify the rules into different categories, typically based on their usage and interrogative forms.

1. **Quantifiers, Adjectives, Adverbs**: Words with the QC, RB, and JJ POS tag, respectively. For words with JJ tags, the word and the corresponding determiners (if present) are selected as the answer candidate.

2. **Possession based**: Words with PRP and NN tags that have suffixes as "టి"," యొక్క", "కి" and "కు" (Ti",yokka",ki" and ku"). The suffix "టి" (Ti") is used for words like "అతని", "వాళ్ళ", "కంటి", "విద్యార్థుల" (atani"-his, vAlla"-their's, kanTi"-eyes', vidyArthula"-students')

3. **Time-Place based** : Noun words with a "లో" (lO") suffix, along with other words present in custom list of time-related words ("మార్నింగ్", "ఇయర్")(morning", year") come under this category.

4. **Direct and Reported Speech**: The word "అని" is generally used to denote direct speech in Telugu. Phrases before the word "అని", along with phrases in quotation marks, are chosen as answer phrases.

5. **Verbs**: Telugu follows the SOV (Subject Object Verb) structure, in general. If the last word has a V" POS tag in a sentence, then we selected the verb and adjacent adverbs as an answer candidate.

6. **Subject**: We use the UD tags to determine the subject of a sentence. As an additional check, we only select the candidate subjects in those sentences whose last word is tagged as the root verb, and the subject is a noun.

6 Question Formation

Questions are formed according to the chosen phrases chosen previously, and the question words are replaced using further conditions if required.

1. **Quantifiers, Adjectives, Adverbs**: The words that are marked JJ POS are replaced with "ఎటువంటి" (eTuvanti"- what kind of) RB POS tagged that are followed by verbs with "గా" (gA") suffix are replaced by "ఎలా" (elA"-how) and the QC tagged words that are not articles ("ఒక" (oka"- one/once) were chosen and changed based on the following word. If the quantifier is followed by "శాతం", "మంది" ,"వరకు" (shAtam",maMdi",varaku") then the word is replaced with "ఎంత" (eMta"-how much), if the quantifier has a suffix it is added to the question word.

   For example: "1700కు" - "ఎంతకు" (eMtaku) and the rest of the quantifiers like ఐదు పిచ్చుకలు (meaning five sparrows) are replaced with "ఎన్ని"(enni"-how many) ("ఎన్ని పిచ్చుకలు" (how many sparrows) in this case).

2. **Possession based**: The nouns and pronouns that satisfied the rules are replaced with "ఎవరి" (evari"-whose) and the dative cases are replaced with "ఎవరికి" (evariki"-to whom). This could be an exception for non-human nouns and pronouns. In the children's stories, most of the nouns are personified, so there were fewer errors than we presumed.

   For example: A sentence with a phrase

[12] https://github.com/UniversalDependencies/UD_Telugu-MTG (Bogdani, 2018)

like "రాముడి ఇల్లు..." (ram’s house...) would form a question like "ఎవరి ఇల్లు.." (whose house..)

3. **Time-Place based**: We made a list of words that are used to convey time. If the lemma of the word matched the word in the dictionary, then we marked it time" and was replaced with "ఎప్పుడు" (eppuDu"-when) or else it was marked as a place and replaced with "ఎక్కడ" (ekkaDa"-where).

   For example: A sentence with the phrase "రేపు వస్తాడు" (he will come tomorrow) will form a question "ఎప్పుడు వస్తాడు?"(when will he come).

4. **Direct and Reported Speech** : The whole speech phrase or the phrase that is quoted is replaced with "ఏమని" (Emani") in the sentence.

   For example: A phrase in quotes in a sentence like దుర్యోధనుడు "ఏమంటివి ఏమంటివి..!" అని అన్నాడు. (Duryodhan said,"what did you say..!".) would form a question like దుర్యోధనుడు ఏమని అన్నాడు? (what did Duryodhan say?)

5. **Verbs** : The verb is replaced with "ఏమి చేస్తూ" Emi cEstu"-doing what) + <suffix>". The appropriate suffix is chosen from the information lost in the lemmatized word.

   Additionally, the verb tags were used to form polar questions. The interrogative form of a sentence in Telugu can be constructed by adding intonation to the verb, so we added "ఆ" (A") vowel at the end of the verb to make a yes or no question. The answer phrase to this question would be "అవును" (avunu"-yes), followed by the original phrase.

   For example: A sentence with a verbal phrase like "సీత వెళుతూ ఉంది"(Sita is going) will form a question like "సీత ఏమి చేస్తూ ఉంది? "(What is Sita doing?).

6. **Subject** : Based on the suffix of the verb the subject is replaced with "ఏది", "ఏవి" or "దేని", "వేటికి" (meaning what, which simultaneously) or "ఎవరు" (evaru"-who) if the subject has a gender and marked a human in POS tags, and the root suffix is changed accordingly for "ఎవరు" (evaru"-who (honorific)).

   For example: "గంగ అక్కడి నుంచి వెళ్లిపోయింది." (Ganga left from that place) forms a question like "ఎవరు అక్కడి నుంచి వెళ్లిపోయారు?" (Who left from there?).

7 Answer Evaluation

User’s answer for the question generated is evaluated in two ways depending on the form of input.

1. Telugu Answer Evaluation
2. Multilingual Answer Evaluation

7.1 Telugu Answer Evaluation

A string input in Telugu is taken from the user and string matching is done for the whole sentence to the answer phrase stored from Question and Answer Pair Generation. Answer could be either in the sentence form or in a phrasal form that has the keywords which the question was formed on.

7.2 Multilingual Answer Evaluation

7.2.1 Sentence Transformers

Similar to word embedding, where the learned representation of same words have similar representation, sentence embedding (Nikhil, 2017) maps semantic information of sentences into vectors. Multilingual Sentence Embedding deals with sentences in multiple languages that are mapped in a closer vector space if they have similar meanings.

Sentence Transformers are Multilingual Sentence Embedding (Ivana Kvapilíková, 2020; Mikel Artetxe, 2019) formed using BERT / RoBERTa / XLM-RoBERTa & Co. with PyTorch[13]. This framework provides an easy way of computing dense vector representation of sentences in multiple languages. They are called sentence transformers since the models are based on transformer networks like BERT / RoBERTa / XLM-RoBERTa etc.

We use a pre-trained sentence transformer (Nils Reimers, 2019) based cross-lingual sentence embedding system which can take a sentence in a language and create an embedding in a multilingual space. The answer phrases and sentences are stored in a dictionary. The answers in a different language are taken as an input and are pro-

[13] (Reimers, 2021)(Horev, 2018) (Ferreira, 2020)

jected into multilingual space and the similarity is checked using cosine similarity with the stored answer phrase in Telugu.

In the final system we used syntax matching to mark the user's answer if the input is in Telugu and used sentence transformers if the input is in any other lanuage.

8 Results

We obtained results that resemble commonly used questions covering nine POS and UD tags. The questions generated by this system are successful and are most similar to academic questions we see in textbooks. We did manual error analysis for the question and answer pair generated. In most cases, it has produced legible results that resemble human-made questions, but there were errors in a few complex sentences. Out of the 916 questions formed, only 34 were either completely erroneous or illegible. The rest were both grammatically correct and significant for the context of the story. The system successfully obtained all possible questions for each simple sentence, not requiring further linguistic analysis.

Table 1 lists the number of times each question word occurred and the number of times it appeared wrong in the experiment with five stories. Table 2 in section 9 shows the sample question and answers generated by the system for children stories.

8.1 Question Generation Error Analysis

The Question Generation by the system is manually annotated by two human evaluators with Computational Linguistics background. Guidelines given to the evaluators are:

- Question with grammatical mistakes are marked as errors.
- Semantic errors in question are marked as errors.
- Questions that are highly irrelevant to the story are marked as errors.

Errors are equally influenced by the word tags, the context of the word, and the word's position in a sentence. We analysed each and every way the errors occurred and could occur.

Errors in elA" ('how') questions are often caused due to spaces between the words and suffixes in the dataset we chose.

Question word	**Occurrences**	**Errors**
ఎలా (elA)	64	2
ఎన్ని (enni)	76	5
ఎంతకు (eMtaku)	4	0
ఎంత (eMta)	3	0
ఎవరి (evari)	187	0
ఎవరికి (evariki)	1	0
ఏమి (Emi)	69	3
దేని (dEni)	45	10
ఎవరు (evaru)	20	0
ఎప్పుడు(eppuDu)	7	0
ఎక్కడ (ekkaDa)	21	5
ఏమి చేస్తూ (Emi cEstU)	148	2
ఏమని (Emani)	10	0
ఆ (A)	148	0
ఎటువంటి (eTuvaMTi)	103	6
వేటికి (vETiki)	10	1

Table 1: Question Types

enni" (quantifier - based) questions are built from diverse quantifiers (for example: time, age, number of people - these quantifiers are often written as sandhi with the word, which causes the POS tagger to give ambiguous tags) and numerous ways of writing quantifiers in Telugu. Few quantifier question word errors occurred due to wrong POS tagging of cross-coded words (words that are actually in English but written in Telugu script). In Telugu, two numbers are used together when representing non-specific quantities between the two numbers (x y means from x to y), for example, reMDu (two) mUDu(three) nimishAlu (minutes)" meaning two to three minutes. This kind of representation makes the system assume there are two quantifiers, and the sentence is eligible for two questions based on the same.

dEni" (subject-based) questions have errors because of ambiguous suffixes and inaccuracies in UD tagging. The lack of human identification in the system made human subjects also replaceable with dEnini" instead of evarini". Another error was due to subjects that were nominal (names) with end syllables similar to common suffixes (which are included as word context in the rule formation). These names were split and formed incorrect question words. For example, the name

Shalini" was converted to interrogative form as dEnini". The rest of the errors are due to wrong POS tags, cross-codes, and initials/abbreviations.

Emi" ('what') question forms also have similar POS tags and cross-codes issues. Few of these errors occurred due to punctuation marks between the same sentence, breaking it up into multiple sentences.

eTuvaMTi" ('what-kind-of') question forms run into issues where there is personification. General questions based on adjectives for humans are based on a person's subtle qualities; however, in a few cases, the adjective that was chosen is inapt to be formed into a question (less similar to human made question). The question that was formed was still grammatically correct in both human and non-human subjects; nevertheless, it is more suitable and precise for a non-human noun. For example (ఎలాంటి శాలిని/what kind of Shalini-పరిచయమైన శాలిని/ the Shalini, that I know)

ekkaDa" ('where') based question forms show errors when an abstract word is used as a place, for example - In thoughts", In that age". Certain quantitative words in Telugu can be appended with -lO to convey meanings like in youth", in hundreds". They tend to pass the rules in question generation. Our list of time-related words is not exhaustive, so a few time-related words are also tagged under ekkaDa" (place) because of the same suffix.

Most of the tags are error free except for a few ambiguous errors since the rules select answer phrases precisely or do not consider it. Some of the examples of the questions that are produced by the system are listed below in Table-2 in the appendix. The results can be improved to make the question formation more precise by increasing the number of rules by observing further data.

The anaphora resolution is a limitation in this system; thus, most of the in-appropriation in the answer section was caused due to this.

For example:

Q: ఎవరి చదువంతా సిటీలో , దర్జాగా ... సాగింది?
Q: Whose studies got completed in the city luxuriously?
A: నీ చదువంతా సిటీలో , దర్జాగా ... సాగింది .
A: Your studies got completed in the city luxuriously.

In this case the question is aptly formed but the answer is slightly ill-formed.

There were few errors due to the POS tagger we used. It marked wrong POS tags for cross coded text.

For example:

Q: నీలం కుమావత్, ఎన్ని?
Q: Neelam Kumawath, how many?
A: నీలం కుమావత్ , ఐ .
A: Neelam Kumawath, I.

The error in this question and answer pair is the "ఐ" 'I' which is an initial (Neelam Kumavat, I) is marked as a number.

9 Conclusions

We have built a mixed rule-based and AI-based question and answer generating system with 96.28% accuracy.

We used two methods for summarization and two similarity measures. We constructed observation-based rules for the dataset in a particular domain. There is a chance of varying results if we test this system for data in a different domain, but it gives accuracies above 95% for any data in the domain chosen.

We tested question generation in the news article domain, which gave grammatically correct questions. The error rate may increase if we use complex words and phrases that need tags beyond the proposed set of rules.

We plan to extend our work to be able to include:

1. Anaphora Resolution
2. Extending to other domains

3. Cover more types of questions
4. Improving the UD tagging model

For testing the meticulousness of the user, as a future task, we wish to use:

1. Questions on minor details
2. NE (Named Entities) and CN (Common Nouns)

Q: ఎటువంటి మోట తో వంగడం కష్టంగా వుంది?
A: అంత పెద్ద మోట తో వంగడం కష్టంగా వుంది

Q: చెప్పులు , బట్టలు , గాజులు , పళ్ళు, గిన్నెలు బజారులో ఎలా కొని , ఊళ్ళో ఇంటింటికి వెళ్లి అమ్ముకునే వాడు?
A: చెప్పులు , బట్టలు , గాజులు , పళ్ళు , గిన్నెలు బజారులో చవకగా కొని , ఊళ్ళో ఇంటింటికి వెళ్లి అమ్ముకునే వాడు

Q: సామాన్లన్నీ మోట కట్టి , గాడిద మీద వేసి , బజారు నుంచి ఊళ్ళో , ఊళ్లోనుంచి తిరిగి ఎవరి ఇంటికి తిప్పేవాడు?
A: సామాన్లన్నీ మోట కట్టి , గాడిద మీద వేసి , బజారు నుంచి ఊళ్ళో , ఊళ్లోనుంచి తిరిగి అతని ఇంటికి తిప్పేవాడు

Q: అమాయక పిచుక ఎక్కడకి, ఎందుకు అని అడగకుండా, ఆ కాకులను గుడ్డిగా నమ్మి ఏమి చేసింది?
A: అమాయక పిచుక ఎక్కడకి, ఎందుకు అని అడగకుండా, ఆ కాకులను గుడ్డిగా నమ్మి వాటితో వెళ్ళింది.

Q: పిచుక మాట నమ్మలేదు కదా , దాని వైపు అసహ్యంగా చూసి మరో ఎన్ని దెబ్బలు వేసారు?
A: పిచుక మాట నమ్మలేదు కదా , దాని వైపు అసహ్యంగా చూసి మరో రెండు దెబ్బలు వేసారు

Q: ఆ కాకులతో పిచుకకి స్నేహం అయ్యిందా?
A: అవును, ఆ కాకులతో పిచుకకి స్నేహం అయ్యింది.

Q: ఒకానొకప్పుడు ఎక్కడ ఒక అమాయకపు పిచుక వుండేది?
A:ఒకానొకప్పుడు ఒక ఊరిలో ఒక అమాయకపు పిచుక వుండేది.

Q: ఏమని పిచుక ప్రాధేయ పడింది?
A:బాబోయ్! బాబోయ్! నా తప్పేమీ లేదు, నేను అమాయకురాలిని, నేనేమీ చేయలేదు, నన్ను వదిలేయండి! అని పిచుక ప్రాధేయ పడింది.

Table 2: Sample questions generated by the system

References

Lucy Vanderwende Ani Nenkova. The impact of frequency on summarization.

Bogdani. 2018. Parts of speech tagging using lstm (long short term memory) neural networks.

Luis Argerich Rosita Wachenchauzer Federico Barrios, Federico Lt'opez. 2016. Variations of the similarity function of textrank for automated summarization.

Diogo Ferreira. 2020. What are sentence embeddings and why are they useful?

Shawkat Guirguis Hafedh Hussein, Mohammed Elmogy. 2014. Automatic english question generation system based on template driven scheme.

Tsukasa Hirashima Akira Takeuchi Hidenobu Kunichika, Tomoki Katayama*. 2004. Automated question generation methodsfor intelligent english learning systems and its evaluation.

Agus Gunawan Holy Lovenia, Felix Limanta. 2018. Automatic question-answer pairs generation from text.

Kyoung-Soo Han Do-Sang Yoon Joo-Young Lee Hae-Chang Rim Hoojung Chung, Young-In Song. A practical qa system in restricted domains.

Rani Horev. 2018. Bert explained: State of the art language model for nlp.

Gorka Labaka-Eneko Agirre Ondej Bojar Ivana Kvapilíková, Mikel Artetxe. 2020. Unsupervised multilingual sentence embeddings for parallel corpus mining.

Prateek Joshi. 2018. An introduction to text summarization using the textrank algorithm (with python implementation).

Xu Liang. 2019. Understand textrank for keyword extraction by python: A scratch implementation by python and spacy to help you understand pagerank and textrank for keyword extraction.

Detmar Meurers Maria Chinkina. 2017. Question generation for language learning: From ensuring texts are read to supporting learning.

Holger Schwenk Mikel Artetxe. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond.

Ms. Vrushali Bhise Rushali A. Deshmukh Mr. Shubham Bhosale, Ms. Diksha Joshi. Automatic text summarization based on frequency count for marathi e-newspaper.

Nishant Nikhil. 2017. Sentence embedding: A literature review - towards data science.

Iryna Gurevych Nils Reimers. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks.

Rajathurai Nishanthi. 2020. Understanding of the importance of mother tongue learning.

Mukul Hase Manish Shrivastava Payal Khullar, Konigari Rachna. Automatic question generation using relative pronouns and adverbs.

Sivaji Bandyopadhyay Rami Reddy Nandi Reddy. 2006. Dialogue based question answering system in telugu.

Nils Reimers. 2021. Sentencetransformers documentation - sentence-bert: Sentence embeddings using siamese bert-networks.

Roshni. 2020. 8 reasons why the nep's move to teaching in mother tongue could transform teaching and learning in india.

Md. Sajjatul Islam Md. Shahnur Azad Chowdhury-Md. Jiabul Hoque Shudipta Sharma, Muhammad Kamal Hossen. Automatic question and answer generation from bengali and english texts.

Serge Sharoff Siva Reddy. 2011. Cross language pos taggers (and other tools) for indian languages: An experiment with kannada using telugu resources.

10 Appendix

List of words related to time:
'అప్పుడు', 'రోజు' , 'కాలం', 'సాయంకాలం', 'ఉదయం', 'మధ్యాహ్నం', 'రాత్రి', 'పగలు', 'నెల', 'వారం', 'సంవత్సరం', 'సూర్యాస్తమయం', 'శుభోదయం', 'దినం', 'సమయం', 'వర్తమానం' , 'పూర్వం', 'భవిష్యత్తు', 'సోమవారం', 'మంగళవారం', 'బుధవారం', 'గురువారం', 'శుక్రవారం','శనివారం','ఆదివారం','మాసం'

Translations Then, day, time period, evening, morning, afternoon, night, morning(synonym), month, week, year, sunset, sunrise, day(synonym), time, present, past, future, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday, month(synonym).

Table 3: This set comprises of the time-related words that have a high chance of being used in a storybook.

Q:What kind of sack was hard to carry?
A:That much of a heavy sack was hard to carry.

Q:In the market how was he buying sandals, clothes, bangles, fruits, utensils - and sold them in the village?
A:In the market how was buying sandals, clothes, bangles, fruits, utensils for cheap rates and sold them in the village.

Q:Packing all the things, putting them on the donkey, from market to village, from village to whose house was he taking them?
A:Packing all the things, putting them on the donkey, from market to village, from village to his own house he was taking them.

Q:How did the innocent sparrow believed the crows without even asking why and where?
A:The innocent sparrow believed the crows blindly without even asking why and where.

Q:Instead of believing the sparrow, looking at it with disgust how many times did they beat it?
A:Instead of believing the sparrow, looking at it with disgust they beat it 2 times.

Q:Did the sparrow made friends with the crows?
A:Yes, the sparrow made friends with the crows.

Q:Once upon a time where was the innocent sparrow living?
A:Once upon a time the innocent sparrow was living in a village.

Q:What did the sparrow say pleadingly?
A:The sparrow said pleadingly, "No! no! I didn't do any mistake, I'm innocent, I did nothing, please leave me."

Table 4: Translations of the results in Table 2 in section 9